Microsoft Azure Data factory

Cognitive Convergence is Subject Matter Expert in Office 365, Dynamics 365, SharePoint, Project Server, Power Platform: Power Apps-Power BI-Power Automate-Power Virtual Agents.
We offer Power BI consulting services covering solution architecture refinement, customization, integration, transformation, visualization and analytics to uncover insights hidden within data and enhance data exploration.

CONTENTS

Objectives....4

What is Azure Data Factory?....4

Azure Data Factory....4

Flow Process of Data Factory....4

Data Integration service....5

Input dataset....5

Pipeline....5

Output dataset....5

Linked services....5

Gateway....6

Cloud....6

Why Azure Data Factory....6

Key Components of Azure Data Factory....6

Collect and connect data....6

Transforming the data....6

Publishing the data....7

Monitoring your data....7

Pipeline....7

Activity....7

Datasets....7

Linked Services....7

Triggers....7

Control flow....8

Azure Data Factory a source of ETL....8

Create Azure Data Factory....8

Create a linked service....9

Create datasets 10

Create a pipeline 12

Debug the pipeline *13*

Trigger the pipeline manually *14*

Copy Data tool 14

Create Azure Data Factory Data Flow 16

Create new data flow *19*

Use cases 19

Easy-to-use *19*

Cost-effective *19*

Powerful *19*

Intelligent *20*

Why use Azure Data Factory? 20

Ability to use Azure Data Factory is a skill in high demand *20*

Frequent updates and features released in Azure Data Factory *20*

Documentation *20*

Easy to get started with *20*

Pay-as-you-go and Scale *20*

Code Version Control *20*

Continuous integration and continuous deployment (CI/CD) *21*

Monitor Alerts *21*

Web Portal *21*

Volume, Variety and Velocity *21*

Platform-as-a-Service (PaaS) *21*

Built-in Connectors and Azure Integration *21*

ETL and ELT *21*

SSIS Investment *21*

Parallel execution *22*

Submit new features to the Product Team *22*

PowerShell *22*

One-stop-shop for Analytics *22*

Security *22*

Advantages of Using Azure Data Factory (ADF) 22

Disadvantages of Azure Data Factory 23

Pricing 23

Data Factory Pipeline Orchestration and Execution *23*

Conclusion 24

Contact us 24

OBJECTIVES

In this paper, we will discuss what Azure Data Factory is and what its use cases are, what are the key components and features of azure data factory, and how it works. Here we will also discuss how to create a data factory, dataset, pipeline, and monitor and trigger pipeline, linked services, copy data tool, creating new data follow.

WHAT IS AZURE DATA FACTORY?

Data generated by several applications of products is increasing exponentially day by day. As the data is coming from many sources, it is very difficult to manage it.

To analyze and store all this data, we can use Data Factory which:

- Stores data with the help of Azure Data Lake Storage
- Analyzes the data
- Transforms the data with the help of pipelines (a logical grouping of activities that together perform a task)
- Publishes the organized data
- Visualizes the data with third-party applications like Apache Spark or Hadoop.

AZURE DATA FACTORY

Azure Data Factory is defined as a data integration service. The aim of Azure Data Factory is to fetch data from one or more data sources and convert them into a format that we process. The data sources might contain noise that we need to filter out. Azure Data Factory connectors enable us to pull the interesting data and remove the rest. It can process and transform the data by using compute services such as Azure Data Lake Analytics, Azure Machine Learning, and Azure HDInsight Hadoop. We can publish the output data to data stores such as Azure Data Lake for Business Intelligence (BI) applications to perform visualization or analytics. For better business decisions, we can organize the raw data into meaningful data stores.

FLOW PROCESS OF DATA FACTORY

Azure Data Factory to ingest data and load the data from a variety of sources into Azure Data Lake Storage. It is the cloud-based ETL service that allows us to create data-driven pipelines for orchestrating data movement and transforming data at scale.

Azure Data Factory is a cloud-based integration service that orchestrates and automates the movement and transformation of data. It works heavily on the stored data.

Data Integration service

Data integration involves the collection of data from one or more sources. Then includes a process where the data may be transformed and cleansed or may be augmented with additional data and prepared. Finally, the combined data is stored in a data platform service that deals with the type of analytics that we want to perform. This process can be automated by ADF in an arrangement known as Extract, Transform, and Load (ETL).

Input dataset

This represents the collection of data within the data stores. The data passes through a pipeline for processing.

Pipeline

A pipeline consists of a group of activities, such as:

- Data movement activity
- Data transformation activity using
    - SQL
    - Stored procedures
    - Hive

Output dataset

After the data is transformed into the pipeline, an output dataset is obtained in a structured form.

Linked services

The data from output datasets passes to linked services, such as:

- Azure Data Lake
- Block storage
- SQL

Linked services contain information needed to connect to external sources.

Gateway

This connects on-premises data to the cloud. It consists of a client agent that is installed on the on-premises data system, which then connects to the Azure data.

Cloud

The data is analyzed and visualized using a number of analytical frameworks, like Apache Spark, R, Hadoop, and so on.

WHY AZURE DATA FACTORY

As the world is moving into the cloud and big data; data integration and migration will remain essential elements for organizations across industries. Azure Data Factory helps to address these two concerns efficiently by enabling to focus on data and allow to schedule, monitor and manage ETL/ELT pipelines with a single view.

The adoption of Azure Data Factory is on the rise because:

- Drive more value
- Better decision-making
- Increase business process agility
- Improves productivity with shorter Time to Market
- Reduce overhead costs
- Integrate data cost-effectively
- Transfer data using pre-built connectors
- Get continuous integration and delivery (CI/CD)

KEY COMPONENTS OF AZURE DATA FACTORY

Azure Data Factory consists of various components. These components will transform source data and make it consumable for the end product

Collect and connect data

Data in the pipeline can be copied or transferred to the cloud source data stores or the same data can also be moved from on-premises.

Transforming the data

Computer services can help you in transforming or processing the data which is already centralized in the cloud. Computer services are Hadoop, Spark, R, etc.

Publishing the data

Data that is already structured, analyzed, and well-refined is now collected into the Azure data factory warehouse. It is monitored and published in the Azure data warehouse.

Monitoring your data

PowerShell, Azure monitor is available on the Azure portal which helps you in pipeline monitoring. Azure data factory works on data-driven workflow with structure so that it can easily move and transfer the data. Azure data factory does not work with a single process. It has various small components which work independently, and when combined, it performs successful operations.

Pipeline

A unit of work that is performed by logical grouping activities is called a pipeline. Pipelines can be single or multiple. Various tasks are performed by pipeline at a time, such as transforming, analyzing, and storing

Activity

Processing steps of the pipelines are represented by the help of Activity. For example - copy activity is widely used to move data from one source to another source.

Datasets

Data sources that are present in the data stores are represented by the data set structures. We can also classify that data in our activities.

Linked Services

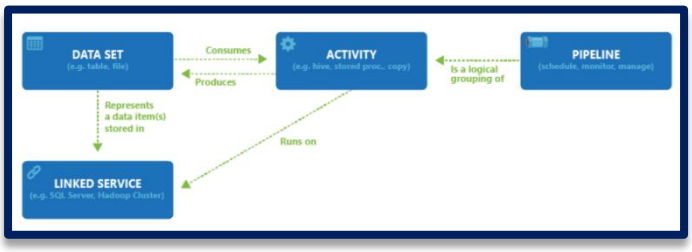
It can be defined as a bridge that is used to connect the Azure data factory to external resources. Computer resources and data stores are types of the linked services.

Triggers

Triggers, as the name suggests, triggers something. When pipeline execution is needed to be disabled then the unit of processing determines it with trigger this unit is known as a trigger. Another feature of the trigger is that we can schedule this process pre-handed so that at a particular point we can trigger and disable the process.

Control flow

It is an extension of activities that are carried out by pipeline. We can also say it is like a thread or chain which arranges activities in a sequence.



AZURE DATA FACTORY A SOURCE OF ETL

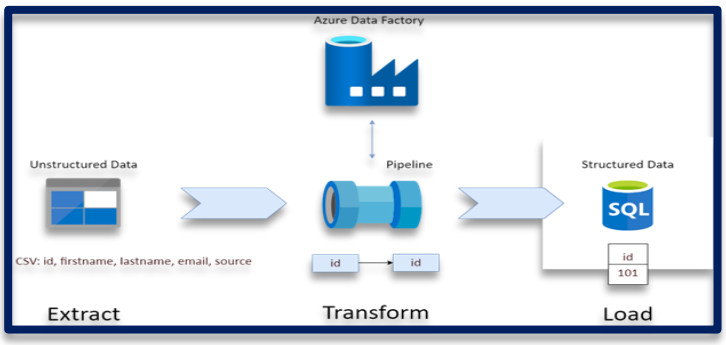
ETL (which stands for extract, transform, and load) is one of the most commonly used paradigms for the data integration process.

The three steps of ETL are:

- Extract: Extracting data from a source location, such as a file or database.
- Transform: Transforming the data from its source format to fit the target location's schema.
- Load: Finally, loading the data into a target location such as a data warehouse for analytics and reporting.

Extract, transform, and load (ETL) is a data pipeline used to collect data from various sources. It then transforms the data according to business rules, and it loads the data into a destination data store. The transformation work in ETL takes place in a specialized engine, and it often involves using staging tables to temporarily hold data as it is being transformed and ultimately loaded to its destination.

The data transformation that takes place usually involves various operations, such as filtering, sorting, aggregating, joining data, cleaning data, reduplicating, and validating data.

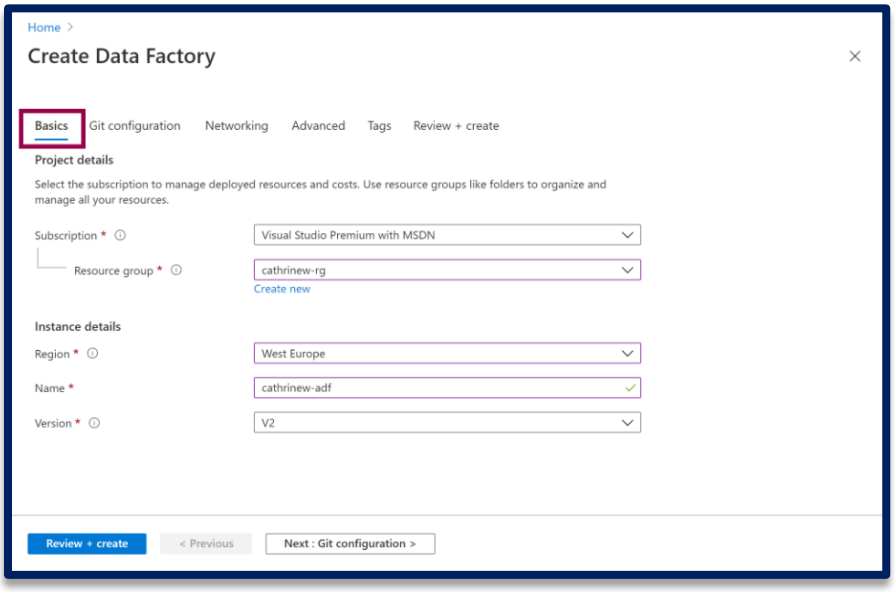


CREATE AZURE DATA FACTORY

In the Azure Portal, open New Data Factory page. Look for “create a resource” or “Data Factories”

- A resource group is a logical container to group resources.
- Choose the closest region

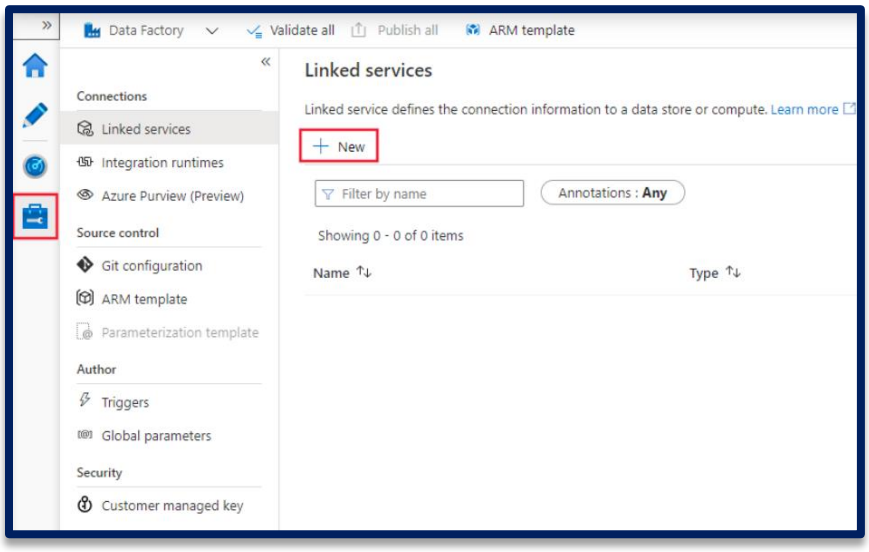
- The name of Azure Data Factory has to be unique across all of Azure.



CREATE A LINKED SERVICE

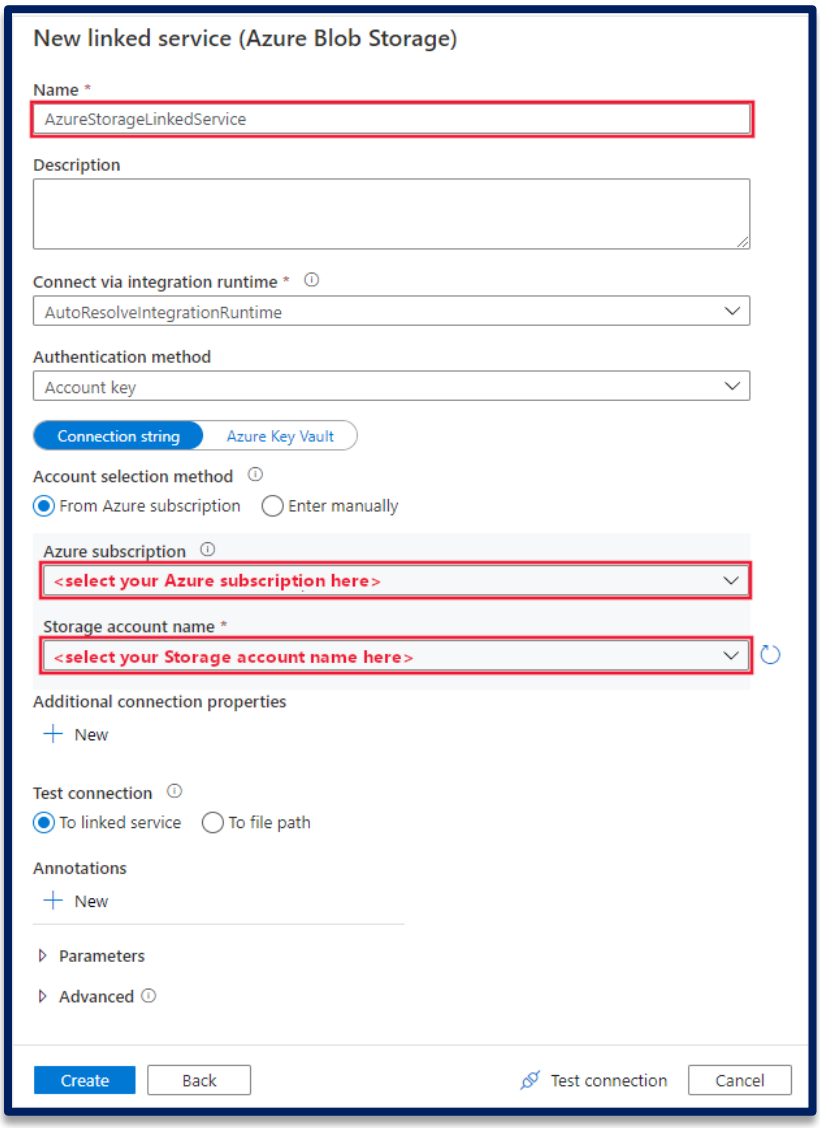
In this procedure, a linked service is created to link Azure Storage account to the data factory. The linked service has the connection information that the Data Factory service uses at runtime to connect to it.

1. On the Azure Data Factory UI page, open Manage tab from the left pane.
2. On the Linked services page, select +New to create a new linked service.



3. On the New Linked Service page, select Azure Blob Storage, and then select Continue.

4. On the New Linked Service (Azure Blob Storage) page, complete the following steps
    a. For Name, enter AzureStorageLinkedService.
    b. For Storage account name, select the name of your Azure Storage account.
    c. Select Test connection to confirm that the Data Factory service can connect to the storage account.
    d. Select Create to save the linked service.



CREATE DATASETS

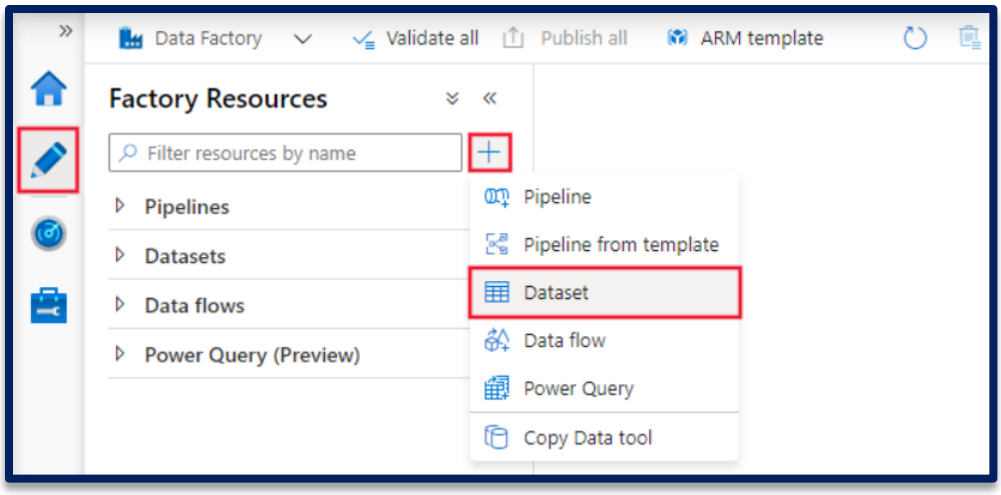
In this procedure, create two datasets: InputDataset and OutputDataset. These datasets are of type AzureBlob. The input dataset represents the source data in the input folder. In the input dataset definition, specify the blob container (adftutorial), the folder (input), and the file (emp.txt) that contain the source data.

The output dataset represents the data that's copied to the destination. In the output dataset definition, specify the blob container (adftutorial), the folder (output), and the file to which the data is copied. Each run of a pipeline has a

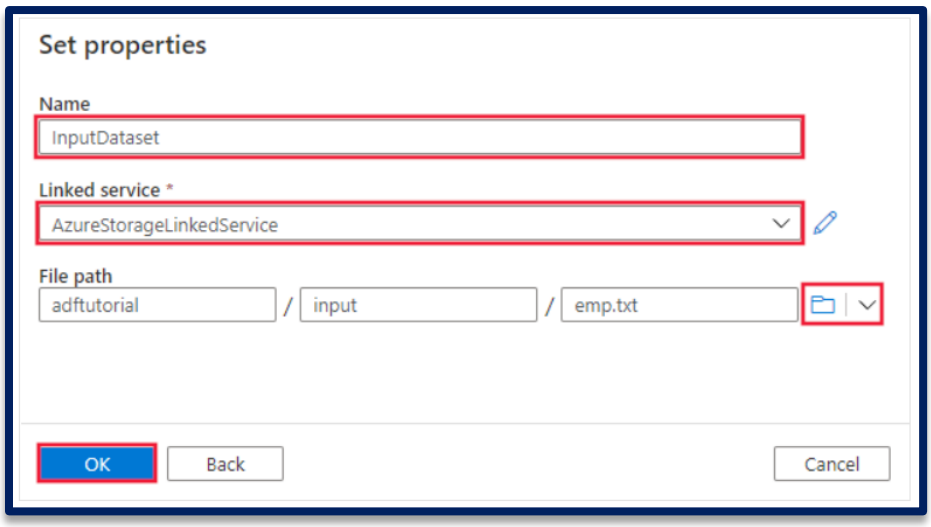
unique ID associated with it. Access this ID by using the system variable RunId. The name of the output file is dynamically evaluated based on the run ID of the pipeline.

In the linked service settings, you specified the Azure Storage account that contains the source data. In the source dataset settings, you specify where exactly the source data resides (blob container, folder, and file). In the sink dataset settings, you specify where the data is copied to (blob container, folder, and file).

1. Select Author tab from the left pane.
2. Select the + (plus) button, and then select Dataset.



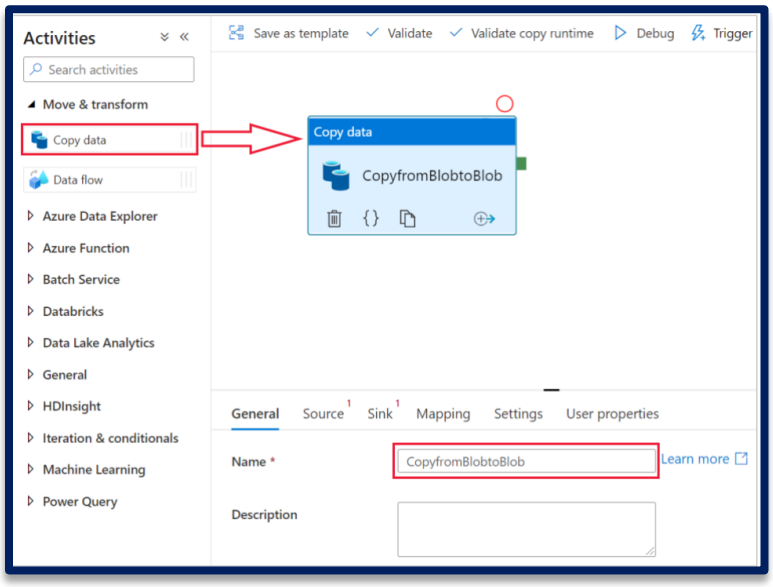
3. On the New Dataset page, select Azure Blob Storage, and then select Continue.
4. On the Select Format page, choose the format type of data, and then select Continue. In this case, select Binary when copy files as-is without parsing the content.
5. Add following properties for input and output data



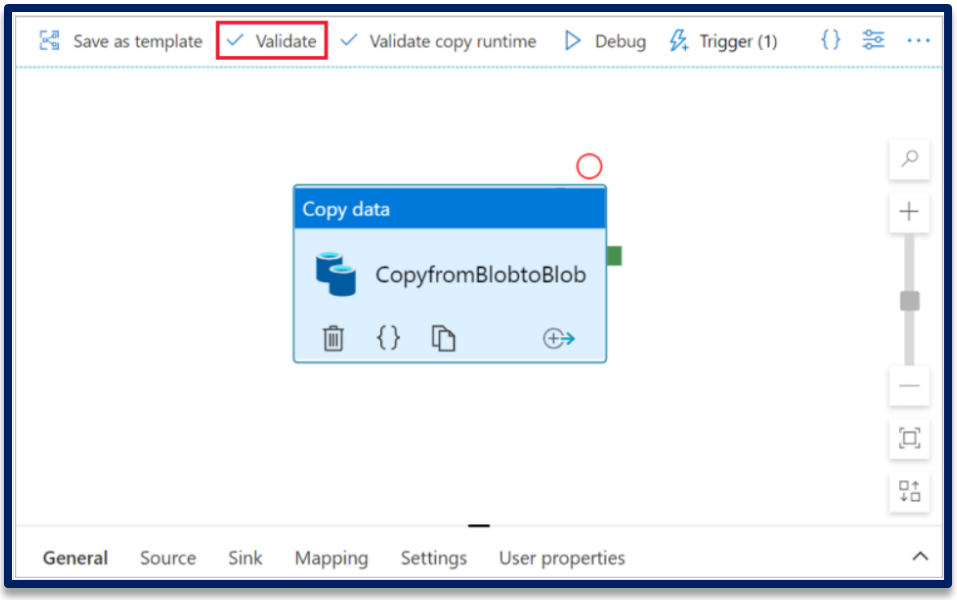
CREATE A PIPELINE

In this procedure, create and validate a pipeline with a copy activity that uses the input and output datasets. The copy activity copies data from the file you specified in the input dataset settings to the file specified in the output dataset settings. If the input dataset specifies only a folder (not the file name), the copy activity copies all the files in the source folder to the destination.

1. Select Pipeline.
2. In the General panel under Properties, specify CopyPipeline for Name. Then collapse the panel by clicking the Properties icon in the top-right corner.
3. In the Activities toolbox, expand Move & Transform. Drag the Copy Data activity from the Activities toolbox to the pipeline designer surface. You can also search for activities in the Activities toolbox. Specify CopyFromBlobToBlob for Name.



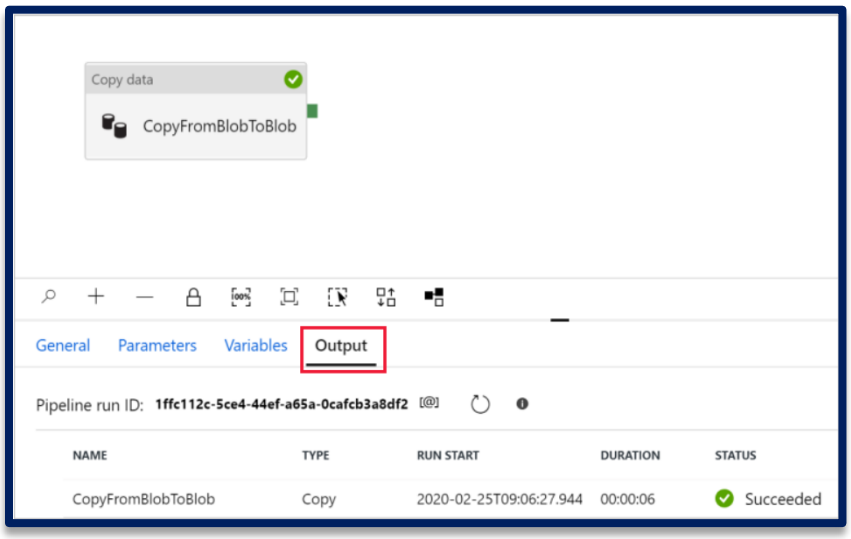
4. Switch to the Source tab in the copy activity settings, and select InputDataset for Source Dataset.
5. Switch to the Sink tab in the copy activity settings, and select OutputDataset for Sink Dataset.
6. Click Validate on the pipeline toolbar above the canvas to validate the pipeline settings. Confirm that the pipeline has been successfully validated. To close the validation output, select the Validation button in the top-right corner.



Debug the pipeline

Debug the pipeline before deploying it to Data Factory.

1. On the pipeline toolbar above the canvas, click Debug to trigger a test run.
2. Confirm that you see the status of the pipeline run on the Output tab of the pipeline settings at the bottom.

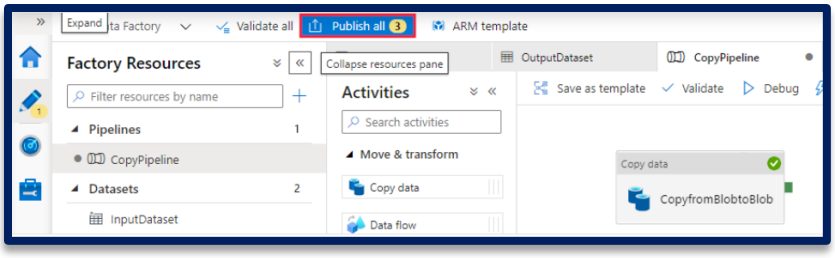


3. Confirm that an output file in the output folder of the adftutorial container. If the output folder doesn't exist, the Data Factory service automatically creates it.

Trigger the pipeline manually

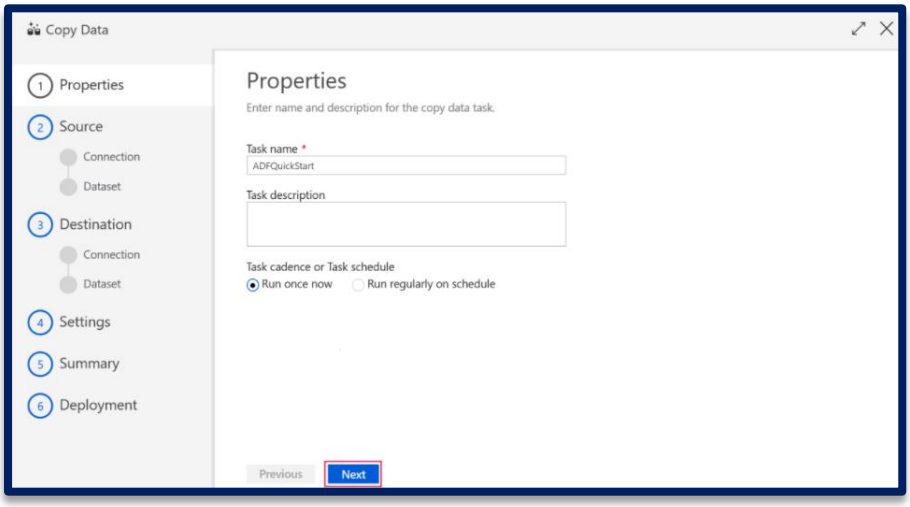
In this procedure, deploy entities (linked services, datasets, pipelines) to Azure Data Factory. Then, manually trigger a pipeline run.

1. Before trigger a pipeline, must publish entities to Data Factory. To publish, select Publish all on the top.
2. To trigger the pipeline manually, select Add Trigger on the pipeline toolbar, and then select Trigger Now. On the Pipeline run page, select OK.

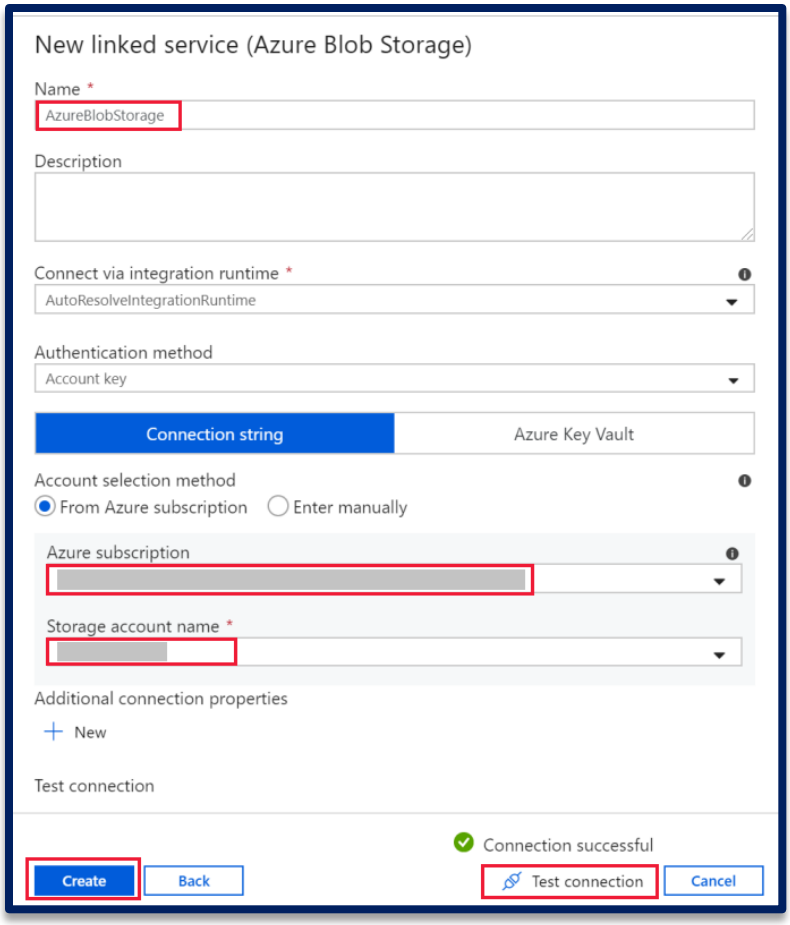


COPY DATA TOOL

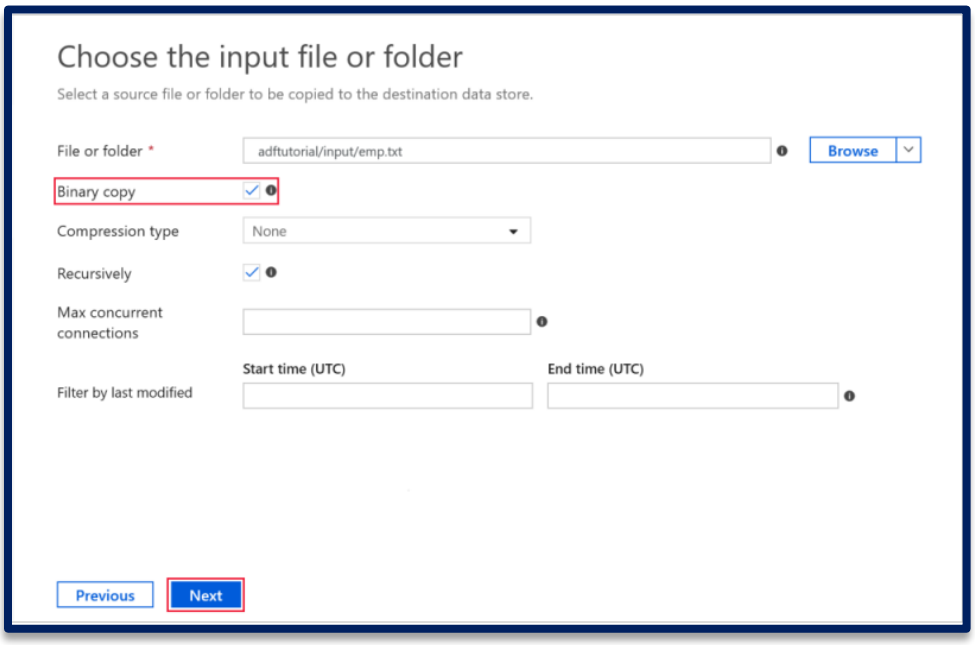
Select the Copy Data tile to start the Copy Data tool. On the Properties page of the Copy Data tool, specify a name for the pipeline and its description, then select Next.



On the Source data store page, create new connection to add a connection, select the linked service type that you want to create for the source connection. On the New Linked Service (Azure Blob Storage) page, specify a name for linked service. Select storage account from the Storage account name list, test connection, and then select Create.

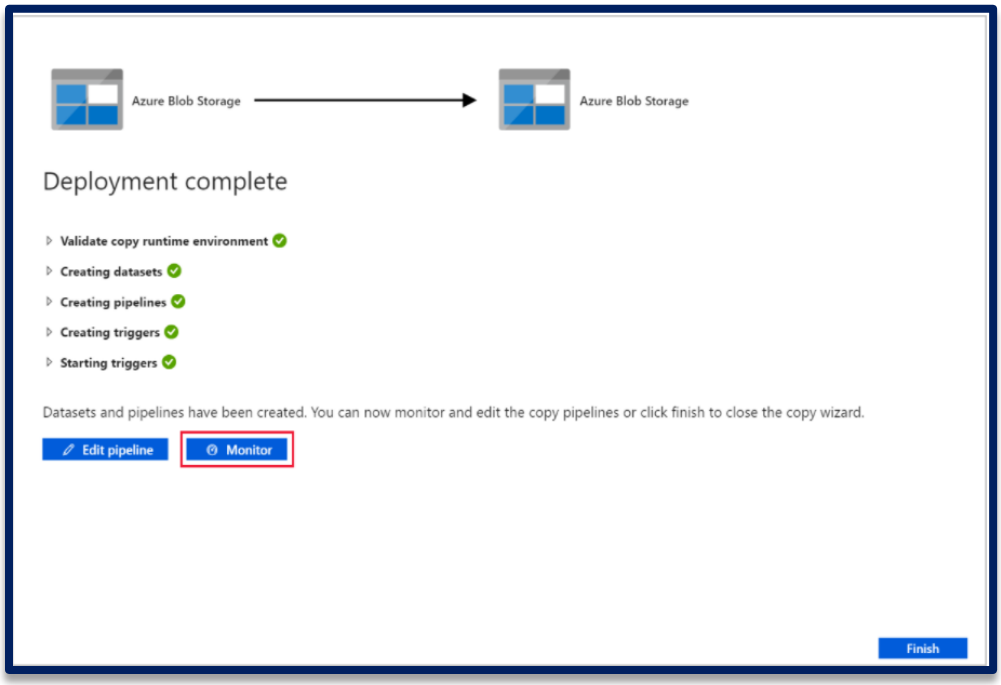


On the Choose the input file or folder page, click Browse to navigate to the input folder, select the emp.txt file, and then click Choose. Select the Binary copy checkbox to copy file as-is, and then select Next.



Similarly select properties for output folder.

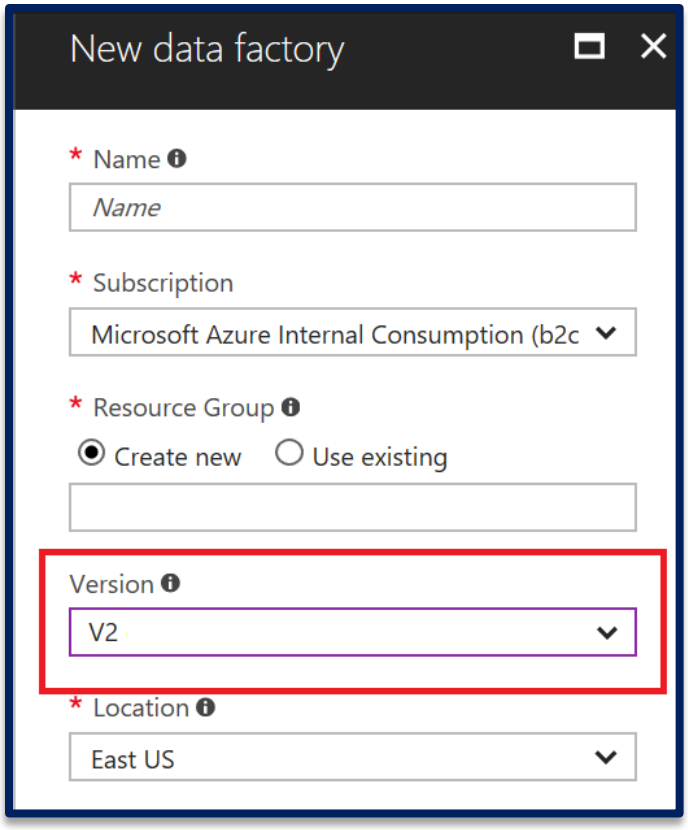
On the Deployment complete page, select Monitor to monitor the pipeline that created.



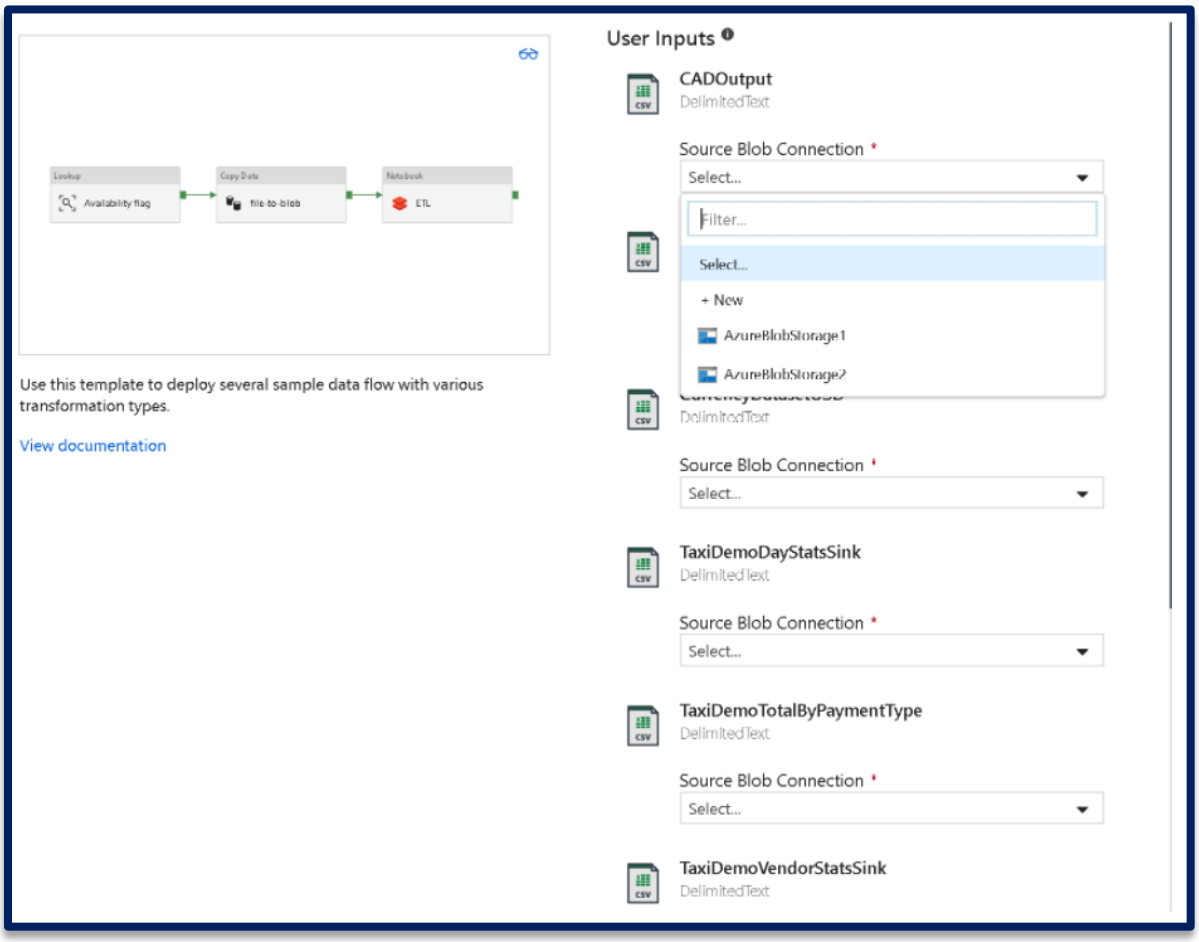
CREATE AZURE DATA FACTORY DATA FLOW

Mapping Data Flows in ADF provide a way to transform data at scale without any coding required. Design a data transformation job in the data flow designer by constructing a series of transformations. Start with any number of source transformations followed by data transformation steps. Then, complete data flow with sink to land results in a destination.

Get started by first creating a new V2 Data Factory from the Azure portal. After creating new factory, click on the "Author & Monitor" tile to launch the Data Factory UI.

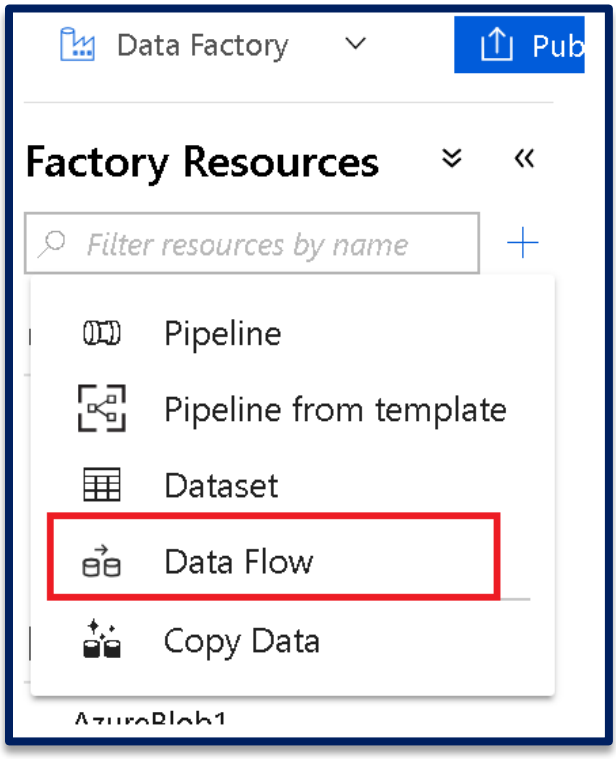


Enter Azure Blob Storage account information after selecting pipeline from template.


Lookup
Availability flag
Copy Data
file-to-blob
Notebook
ETL
Use this template to deploy several sample data flow with various transformation types.
View documentation
User Inputs
CADOutput
DelimitedText
Source Blob Connection *
Select...
Filter...
Select...
+ New
AzureBlobStorage1
AzureBlobStorage2
DelimitedText
Source Blob Connection *
Select...
TaxiDemoDayStatsSink
DelimitedText
Source Blob Connection *
Select...
TaxiDemoTotalByPaymentType
DelimitedText
Source Blob Connection *
Select...
TaxiDemoVendorStatsSink
DelimitedText

Create new data flow

Use the Create Resource "plus sign" button in the ADF UI to create Data Flows.



USE CASES

Easy-to-use

Rehost SQL Server Integration Services (SSIS) in a few clicks and build ETL and ELT pipelines code-free, with built-in Git and CI/CD support.

Cost-effective

Enjoy a pay-as-you-go, fully managed serverless cloud service that scales on demand.

Powerful

Ingest all your on-premises, and software as a service (SaaS) data with more than 90 built-in connectors. Orchestrate and monitor at scale.

Intelligent

Use autonomous ETL to unlock operational efficiencies and enable citizen integrators

WHY USE AZURE DATA FACTORY?

Ability to use Azure Data Factory is a skill in high demand

The rise of cloud technologies and need for data tools helped Azure Data Factory grow quickly. The ability to use Azure Data Factory is already a highly demanded skill in the data analytics space, and it will continue to stay the same. With the rise of skilled users, the process for releasing new ADF features will also increase.

Frequent updates and features released in Azure Data Factory

Microsoft releases new features every month within the service. Likewise, there are monthly updates for previous features.

Documentation

Documentation has greatly improved during the past few years. There are a lot of resources are available through Microsoft documentation and additional information provided by the community.

Easy to get started with

Easily get started with ADF with thanks to great tutorials available in Microsoft documentation and Learning Paths. There is a large amount of community written documentation as well. Beginners as well as experts can navigate the user interface. For example, you can build solutions using the Data Copy Wizard or by dragging and dropping components. This makes it accessible to everyone.

Pay-as-you-go and Scale

Additionally, there is no requirement for an additional monthly or annual license or up-front payments. Simply pay for what you use. Also scale up and down data flows, SSIS nodes, Data Integration Units, etc. when workloads require more or fewer resources.

Code Version Control

Full integration with code version control Git / Github. Nobody can make up any more excuses not to use it.

Continuous integration and continuous deployment (CI/CD)

Full integration with Azure DevOps for CI/CD across different environments and it is easy to setup.

Monitor Alerts

Full integration with Azure Monitor Alerts and Log Analytics. There is no need to build additional alerting frameworks

Web Portal

Design and develop from the web portal, without provisioning additional workstations or servers.

Volume, Variety and Velocity

The 3 Vs for data integration are covered in Azure Data Factory. It doesn't matter how big datasets are, what their structure is, or how fast to load them. Azure Data Factory provides tools and is integrated with other Azure Services to cover any scenario.

Platform-as-a-Service (PaaS)

Azure manages the ADF platform and the backend. It's part of the Platform-as-a-Service (PaaS) offering. Don't need to manage any hardware/middleware or patching.

Built-in Connectors and Azure Integration

There are more than 85 connectors and the number continues to keep growing. This includes some generic connectors like HTTP or ODBC in case can't find a built-in connector.

Azure Data Factory is also fully integrated with other Azure services like Azure Functions, Azure Databricks, Azure Data Explorer, Azure Data Lake and more.

ETL and ELT

Perform ETL (Extract-Transform-Load) using data flows or ELT (Extract-Load-Transform) using only common pipeline activities.

SSIS Investment

In order to made a large investment in SQL Server Integration (SSIS) solutions and they work, don't need to throw away that investment. Take advantage of ADF to execute packages in a managed environment.

Parallel execution

The execution in parallel of activities is built-in by default. So, if trying to load information for many data assets at the same time, achieve this easily without building big frameworks.

Submit new features to the Product Team

As with many other Azure Products, if missing a feature, can submit an idea here. If it is popular and gets voted up, it will be included in the backlog of upcoming features for ADF. It may take a while

PowerShell

A large set of PowerShell APIs is available for use here.

One-stop-shop for Analytics

Azure Data Factory is included as part of the Azure Synapse Analytics Workspace Experience. The service has been called a one-stop-shop for analytics.

Security

- Azure Data Factory offers different security levels and tools to guarantee security, including:
- Data encrypted in transit
- Connections are encrypted
- Role-based access control (RBAC) groups are available in Azure
- Private endpoints
- Use your VNet to host the service
- Governance through Azure Portal

ADVANTAGES OF USING AZURE DATA FACTORY (ADF)

ADF can be used as any traditional ETL tool, but the primary objective is to migrate your data to Azure Data Services for further processing or visualization. A cloud based data integration service that allows organizations in operationalizing, building, debugging, deploying and monitoring the company's big data pipelines. Let us look at the advantages of Azure Data Factory offers:

- It offers a drag and drop interface. You can use it to iteratively build, debug, deploy, and operationalize your big data pipelines. Moreover, it is a codeless workflow management tool for orchestrating data.
- ADF has integration with Azure Services
- With ADF, companies can ingest data from disparate data sources and can turn data into meaningful insights by using BI applications such as Power BI.

DISADVANTAGES OF AZURE DATA FACTORY

Limitations in the service will eventually disappear, but some of them will stay for a longer time.

- Limitations on the number of roles to manage the platform. It's necessary to create a few custom roles.
- Lack of advanced configuration and development options. These limitations arise only in really advanced and complex scenarios. Because solving these issues is not as popular as other features, it can take some time for the Product Team to address these changes.
- Complex pricing model. ADF is one of the most difficult services in Azure for planning an on-going cost and allocating a budget. New features have been released during the past few months to make it easier to understand pricing, but I think that it's still difficult to understand for new users.
- Lack of unit testing and integration with other tools in the Visual Studio stack. By only using the web browser experience, you will run into limitations.
- By using the web browser, you have to make sure to create Azure policies so people cannot access it from their homes or phone using public internet.
- Expressions in data flows or pipelines take some time to understand. At the end of the day, you will always use the same ones, but it takes some time to get to used to them.

PRICING

Pricing for Data Pipeline is calculated based on:

- Pipeline orchestration and execution
- Data flow execution and debugging
- Number of Data Factory operations such as create pipelines and pipeline monitoring

Data Factory Pipeline Orchestration and Execution

Pipelines are control flows of discrete steps referred to as activities. Pay for data pipeline orchestration by activity run and activity execution by integration runtime hours. The integration runtime, which is server less in Azure and self-hosted in hybrid scenarios, provides the compute resources used to execute the activities in a pipeline. Integration runtime charges are prorated by the minute and rounded up. For example, the Azure Data Factory copy activity can move data across various data stores in a secure, reliable, performant, and scalable way. As data volume or throughput needs grow, the integration runtime can scale out to meet those needs.

Type	Azure Integration Runtime Price	Azure Managed Vnet Integration Runtime Price	Self-Hosted Integration Runtime Price
Orchestration	$1 per 1,000 runs	$1 per 1,000 runs	$1.50 per 1,000 runs
Data movement Activity	$0.25/DIU-hour	$0.25/DIU-hour	$0.10/hour

Pipeline Activity	$0.005/hour	$1/hour (Up to 50 concurrent pipeline activities)	$0.002/hour
External Pipeline Activity	$0.00025/hour	$1/hour (Up to 800 concurrent pipeline activities)	$0.0001/hour

CONCLUSION

In this paper we discussed a brief introduction to the data factory and how to use it in Azure Eco-system. Moreover, the benefits of using Azure Data Factory and demoed the steps for creating an instance for the data factory are written here. Here we will also discuss how to create data factory, dataset, pipeline, and monitor and trigger pipeline, linked services, copy data tool, creating new data follow.

CONTACT US

Shahzad Sarwar
Entrepreneur/Architect/Consultant
Cognitive Convergence
http://www.cognitiveconvergence.com
Voice: +1 4242530744
Skype: Shahzad.Sarwar.Online
shahzad@cognitiveconvergence.com